International Journal of Information and Computer Science, 2012, 1: 115-120
Published Online August 2012

www.iji-cs.org

Programmable Logic Controller

D. Ahuja1, N. Chaudhary1
[1]Department of Electrical and Electronics Engineering, YMCA University of Science and Technology, Faridabad, 121006, India

(Abstract)A PLC (Programmable Logic Controllers) is a digitally operating electronic apparatus which uses a programmable memory for the internal storage of instructions by implementing specific functions such as logic sequencing, timing, counting, and arithmetic to control, through digital or analog input/output modules, various types of machines or processes". It is an easily understood programming language .It can hold data for a long time i.e. for a indefinite time but requires excessive work in connecting wires. PLCs are used in many industries and machines. Unlike general-purpose computers, the PLC is designed for multiple inputs and output arrangements, extended temperature ranges, immunity to electrical noise, and resistance to vibration and impact. Programs to control machine operation are typically stored in battery-backed-up or non-volatile memory. A PLC is an example of a hard real time system since output results must be produced in response to input conditions within a bounded time, otherwise unintended operation will result.

Keywords: PLC, Central Processing Unit, NEMA, Sinking, Sources

1. What Does 'PLC' Mean?

A PLC (Programmable Logic Controllers) is an industrial computer used to monitor inputs and based on its program or logic, to control (turn on/off) its outputs to automate a machine or a process. It can also be defined as **"**a digitally operating electronic apparatus which uses a programmable memory for the internal storage of instructions by implementing specific functions such as logic sequencing, timing and counting through digital or analog input/output modules and arithmetic to control"[1].

1.1 History

PLC development began in 1968 in response to a request from an US car manufacturer and were first installed in the industry in 1969.These could be used in the 70′s to send and receive varying voltages and entered in the analog world along with communications abilities. In 80′s there was an attempt to standardize communications with manufacturing automation protocol (MAP), reduction in the size of the PL and making them software programmable through symbolic programming on personal computers instead of dedicated programming terminals or handheld programmers. In the 90′s there was a gradual reduction in the introduction of new protocols and the modernization of the physical layers of some of the more popular protocols that survived the 1980′s [2].The latest standard "IEC 1131-3" has tried to merge PLC programming languages under one international standard. We now have PLCs that are programmable in function block diagrams, instruction lists, C and structured text all at the
same time.

1.2 Traditional PLC Applications

Following are the applications of Traditional PLC:

(i) In automated system, PLC controller is usually the central part of a process control system.
(ii) To run more complex processes it is possible to connect more PLC controllers to a central computer.
Disadvantages of PLC Control
(i)Too much work is required in connecting wires.
(ii) It is difficulty to make changes or replacements.
(iii) It is difficult in finding errors and requires skillful work force.
(iv) When a problem occurs, hold-up time is indefinite
Advantages of PLC Control
(i) It is rugged and designed to withstand vibrations, temperature, humidity and noise.
(ii) It has interfacing for inputs and outputs already inside the controller.
(ii) It can be easily programmed and have an easily understood programming language.

2. PLC Hardware

2.1 Hardware Components of a PLC System

PLC has following hardware components:

(i) Central Processing Unit (CPU)
CPU makes a great number of check-ups of the PLC controller itself so as eventual errors would be discovered early.

Microprocessor based CPU may allow arithmetic operations, logic operators, block memory moves, computer interface, local area network, functions, etc.

(ii)System Buses

The internal paths along which the digital signals flow within the PLC are called buses.The system has four buses:

The CPU uses the (a)data bus for sending data between the different elements, (b) address bus to send the addresses of locations for accessing stored data,(c) control bus for signals relating to internal control actions and (d) system bus for communications between the I/O ports and the I/O unit.

(iii)Memory

System (ROM) is the component to give permanent storage for the operating system and the fixed data used by the CPU.RAM is for data where information is stored on the status of input and output devices and the values of timers and counters and other internal devices. EPROM is for ROM's that can be programmed and then the program is made permanent.

(iv) I/O Sections

It includes inputs monitor field devices, such as switches and sensors and outputs control with other devices, such as motors, pumps, solenoid valves, and lights.

(v)Power Supply

Most PLC controllers work either at 24 VDC or 220 VAC. Some PLC controllers have electrical supply as a separate module, while small and medium series already contain the supply module.

(vi) Programming Device

The programming device is used to enter the required program into the memory of the processor.The program is developed in the programming device and then transferred to the memory unit of the PLC.

3. PLC Operation (Fig.1)

PLC operation takes place through:

(i)Input Relays

Input relays are transistors that connect to the outside world and receive signals from switches, sensors, etc.

(ii)Internal Utility Relays

Internal Utility Relays do not receive signals from the outside world and do not physically exist. These are simulated relays that enables a PLC to eliminate external relays and are dedicated to performing only one task.

(iii)Counters

These are used for a programme to count pulses. They do not physically exist and simulated in nature .They

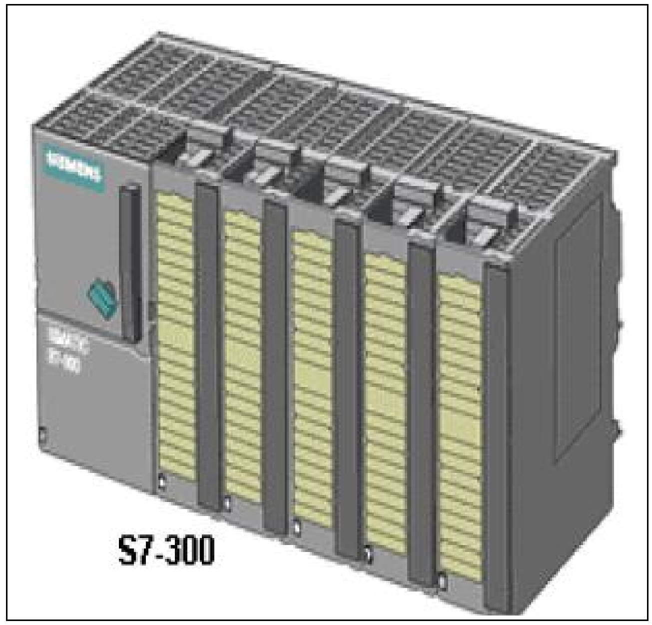


Fig.1 S7-300

can count up, down or both up and down. Since they are simulated they are limited in their counting speed.

Some manufacturers use hardware based highspeed counters .

(iv)Timers

Timers also do not physically exist. These are of many varieties and increments.The most common type is an on-delay type.Others include off-delay and both retentive and non-retentive types. Increments vary from 1ms through 1s.

(v) Output Relays

These are connected to the outside world. They physically exist and send on/off signals to solenoids, lights, etc.

They can be transistors, relays, or triacs depending upon the model chosen.

(vi) Data Storage

Typically there are registers assigned to simply store data .temporarily for math or data manipulation.They can be used to store data when power is removed from the PLC.

4. PLC Programming

4.1 Programming Languages

Programming languages include loading of a program into PLC systems in machine code and a sequence of binary code numbers to represent the program instructions.

A computer program called an assembler can be used to translate the mnemonics into machine code. High level Languages used are C, BASIC, etc.

4.2 Programming Devices

PLC can be reprogrammed through an appropriate programming device given below:

(i)Programming Console
(ii)PC
(iii) Hand Programmer

5. Introduction to Ladder Logic

Ladder logic uses graphic symbols similar to relay schematic circuit diagrams. In ladder diagram two vertical lines represent the power rails and circuits are connected as horizontal lines between these two verticals.

5.1 Ladder Diagram Features (Flg.2)

Following are the features of Ladder Diagram:

(i) Power flows from left to right.
(ii)Output on right side can not be connected directly with left side.
(iii) Contact can not be placed on the right of output.
(iv) Each rung contains one output at least.
(v) Each output can be used only once in the program.
(vi) A particular input a/o output can appear in more than one rung of a ladder.
(vii)The inputs a/o outputs are all identified by their addresses, the notation used depending on the PLC manufacturer.

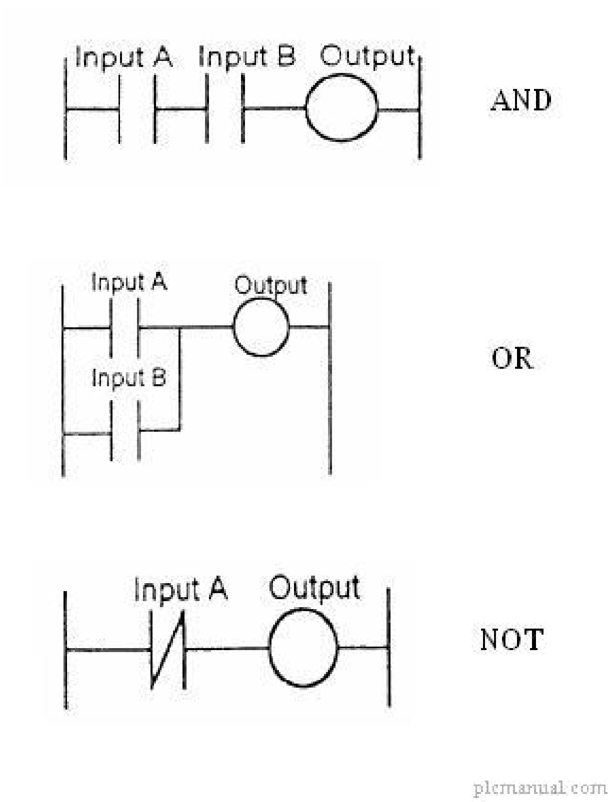


Fig.2 Ladder Diagram

6. Introduction to Statement list

Statement list is a programming language using mnemonic abbreviations of Boolean Logic operations. Boolean operations work on combination of variables that are true or false.

6.1 Statement List Operations

These are following

(i)Load (LD) instruction.
(ii)And (A) instruction.
(iii)Or (O) instruction.
(iv)Output (=) instruction.

7. PLC Installing & Commissioning

PLC Installation, Commissioning and Recommendations (Fig..3)

7.1 Typical Installation

Typical installation includes enclosure, disconnect device, fused isolation transformer, master control relay, terminal blocks ,wir0069ng ducts and suppression devices. Here spacing controllers – follow the recommended minimum spacing to allow the convection cooling. It prevents excessive heat (0–60 C) loss , provides grounding guidelines., takes power , safety and maintenance considerations [4].

7.2 Commissioning and Testing of a PLC System

It includes checking whether:
(i) all cable connections between the PLC and the plant are complete, safe, and to the required specification and meeting local standards.
(ii) all the incoming power supply matches the voltage setting for which the PLC is set.
(iii) all protective devices are set to their appropriate trip settings.
(iv) emergency stop button work.
(v) all input/output devices are connected to the correct input/output points and giving the correct signals.
Loading and testing the software.

7.2.1 Testing Inputs and Outputs

Input devices can be manipulated to give the open and closed contact conditions and the corresponding LED on the input module observed. This involves software, rather than mechanical switching on or off, being used with instructions to turn off or on inputs/outputs.

7.2.2 Testing Software

Most PLCs contain some software checking program through the installed program and provides a list on a screen or as printout with any errors detected.

8. Sinking Sources I/O

Sinking" and "Sourcing" terms are applied only for DC modules.
Sinking = Common GND line (-)
Sourcing = Common VCC line (+)
Most commonly used DC module options in PLCs are:
Sinking input module
Sourcing output module

Sinking I/O circuits on the I/O modules receive (sink) current from sourcing field devices. Sinking output modules are used for interfacing with electronic equipment.
Sourcing I/O: Sourcing output modules are used for interfacing with solenoids.

PLC AC I/O circuits accommodate either sinking or sourcing field devices. Solid-state DC I/O circuits require that they are used in a specific sinking or sourcing circuit depending on the internal circuitry [3].PLC contact (relay) output circuits AC or DC and accommodates either of sinking or sourcing field devices.

9. PLC Applications

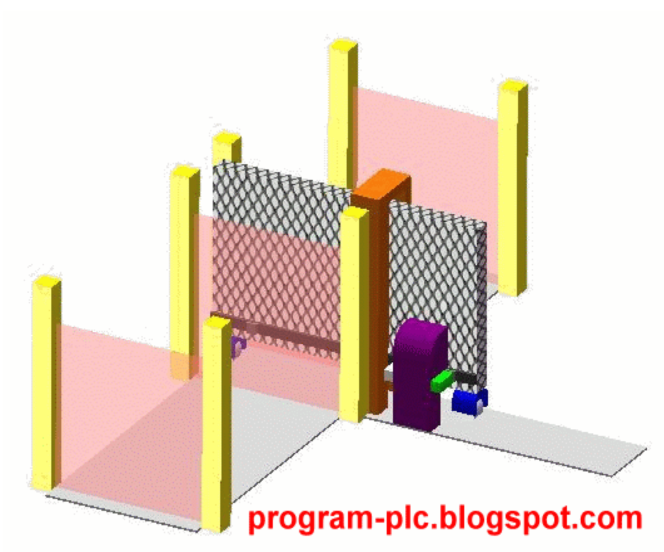


Fig.3 Simulation automatic gates

Following is the information on drawing numbers for automatic gates using PLC (Fig.4):
1. Area Sensor (If there are objects, the sensor output OFF)
2. Area Sensor (If there are objects, the sensor output OFF)
3. Area Sensor (If there are objects, the sensor output OFF)
4. Area Sensor (If there are objects, the sensor output OFF)
5. Electric Motor and Gearbox
6. Limit Switch for Open gate condition
7. Limit Switch for Closed gate conditions
8. Gate

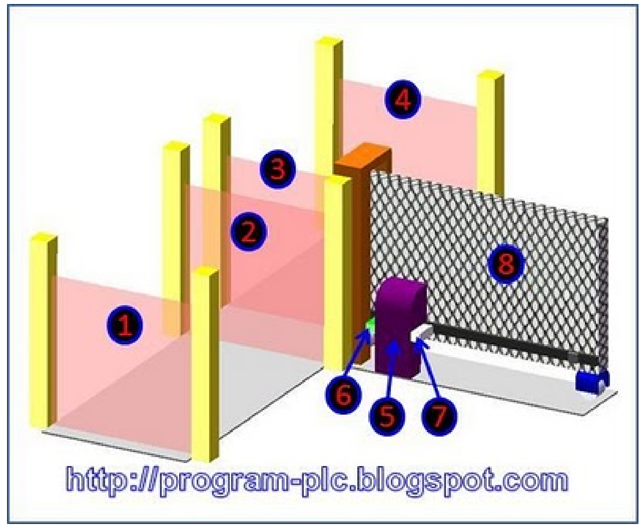


Fig.4 Detail automatic gates using PLC

9.1 Number of Inputs and Output PLC Applied (Fig.5)

9.1.1 Number of Inputs PLC is 6 Input

4 Unit Input for Area Sensor 1, 2, 3, and 4.
1 Unit Input for Limit Switch for Open gate.
1 Unit Input for Limit Switch for Closed gate.
Total Number Of Inputs PLC is Minimum 6 Input Unit.

9.1.2 Number of Output PLC is 2 Output

1 Unit Output to contactor for Electric Motor (Open Gate).
1 Unit Output to contactor for Electric Motor (Close Gate).
Total Number Of Outputs PLC is Minimum 2 Output Unit.

10. Sequence PLC Programming for Automatic Gates (Fig.6)

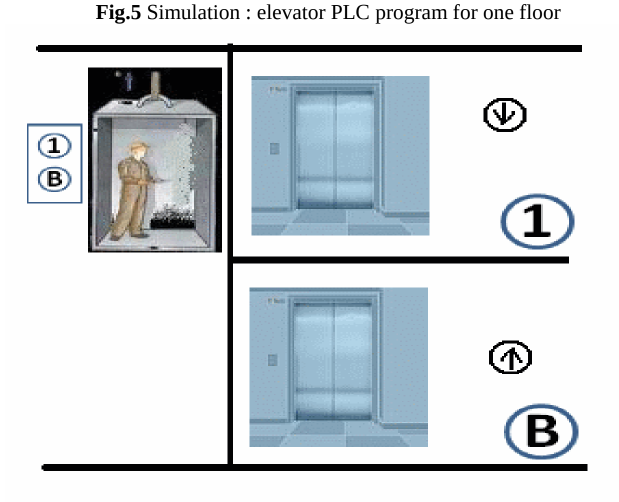
10.1 Open Gate

a. If Area Sensor 1 = OFF Then Electric Motor for Open Gate = ON.
b. If Limit Switch for Open gate = ON Then Electric Motor for Open Gate = OFF.
c. Electric Motor for Close Gate = always OFF

10.2 Close Gate

a. If Area Sensor 4 = OFF AND Area Sensor 2 = ON AND Area Sensor 3 = ON Then Electric Motor for Close Gate = ON.
b. If Electric Motor for Close Gate = ON AND Area Sensor 2 = OFF OR Area Sensor 3 = OFF Then Electric Motor for Close Gate = OFF AND Electric Motor for Open Gate = ON.

Fig.5 Simulation : elevator PLC program for one floor



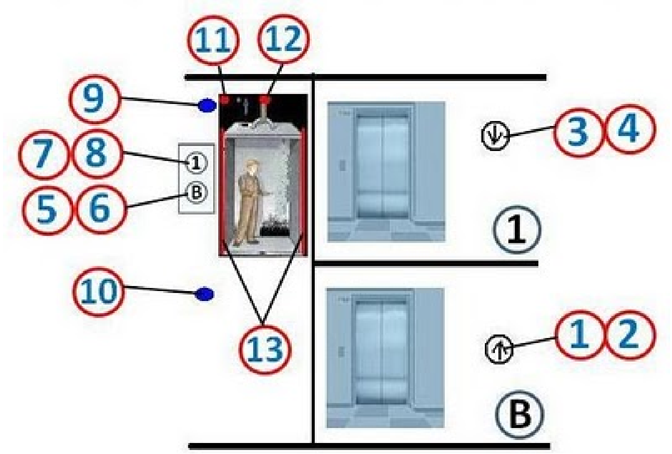


Fig.6 Detail elevator with PLC

Information on Drawing Numbers for Elevator PLC Program (Fig.6)
(1) Push Button Switch (Push Button Switch with Light) In Floor B Outside.
(2) Light Push Button Switch (Push Button Switch with Light) In Floor B Outside.
(3) Push Button Switch (Push Button Switch with Light) In Floor 1 Outside.
(4) Light Push Button Switch (Push Button Switch with Light) In Floor 1 Outside.
(5) Push Button Switch (Push Button Switch with Light) In B Inside.
(6) Light Push Button Switch (Push Button Switch with Light) In B Inside.
(7) Push Button Switch (Push Button Switch with Light) In 1 Inside.
(8) Light Push Button Switch (Push Button Switch with Light) In 1 Inside.
(9) Limit Switch for Position 1
(10) Limit Switch for Position B
(11) Limit Switch for Door Open
(12) Limit Switch for Door Close
(13) Safety Sensor

11. Number of Inputs and Output PLC Applied

11.1. Number Of Inputs PLC is 9 Input

1 Unit Input for Push Button Switch In Floor B Outside.
1 Unit Input for Push Button Switch In Floor B Inside.
1 Unit Input for Push Button Switch In Floor 1 Outside.
1 Unit Input for Push Button Switch In Floor 1 Inside.
1 Unit Input for Limit Switch Position 1.
1 Unit Input for Limit Switch Position B.
1 Unit Input for Limit Switch Door Open.
1 Unit Input for Limit Switch Door Close.
1 Unit Input for Safety Sensor.
Total Number Of Inputs PLC is Minimum 9 Input Unit.

11.2 Number Of Output PLC is 8 Output

1 Unit Output for Electric Motors (Move To Position B)
1 Unit Output for Electric Motors (Move To Position 1)
1 Unit Output for Move the Door Open
1 Unit Output for Move the Door Close
1 Unit Output for Light Push Button Switch in B Inside
1 Unit Output for Light Push Button Switch in 1 Inside
1 Unit Output for Light Push Button Switch in B Outside
1 Unit Output for Light Push Button Switch in 1 Outside
Total Number Of Outputs PLC is Minimum 8 Output Unit.

12. Sequence PLC Programming for Elevator PLC Program

Step 1 : Elevator Up - Down (1 - B)
a. If Push Button B Outside = ON And the elevator was not in a position B Then elevator Move to B.
b. If Push Button B Inside = ON And the elevator was not in a position B Then elevator Move to B.
c. If Push Button 1 Outside = ON And the elevator was not in a position 1 Then elevator Move to 1.
If Push Button 1 Inside = ON And the elevator was not in a position 1 Then elevator Move to 1.
Step 2 : Door Open – Close
a. IF Limit Switch Position B = ON (Pulse / diff.Up) Or Limit Switch Position 1 = ON (Pulse / diff.Up) then Door Open.
b. IF Limit Switch Door Open = ON Or Safety Sensor = ON then Door Close.
c. Continuously to Step 1

PLCs have similar functionality as Remote Terminal Units. An RTU, however, usually does not support control algorithms or control loops. As hardware rapidly becomes more powerful and cheaper, RTUs, PLCs and DCSs are increasingly begin-

ning to overlap in responsibilities, and many vendors sell RTUs with PLC-like features and vice versa. The industry has standardized on the IEC 61131-3 functional block language for creating programs to run on RTUs and PLCs, although nearly all vendors also offer proprietary alternatives and associated development environments [5].

In recent years "Safety" PLCs have started to become popular, either as standalone models (Pilz PNOZ Multi, Sick etc.) or as functionality and safety-rated hardware added to existing controller architectures (Allen Bradley Guardlogix, Siemens F-series etc.). These differ from conventional PLC types as being suitable for use in safety-critical applications for which PLCs have traditionally been supplemented with hard-wired safety relays. For example, a Safety PLC might be used to control access to a robot cell with trapped-key access, or perhaps to manage the shutdown response to an emergency stop on a conveyor production line. Such PLCs typically have a restricted regular instruction set augmented with safety-specific instructions designed to interface with emergency stops, light screens and so forth. The flexibility that such systems offer has resulted in rapid growth of demand for these controllers [6].

13. Conclusion

PLC is an easily understood programming language .It can hold data for a long time i.e. for a indefinite time but requires excessive work in connecting wires. The functionality of the PLC has evolved over the years to include sequential relay control, motion control, process control, distributed control systems and networking. The data handling, storage, processing power and communication capabilities of some modern PLCs are approximately equivalent to desktop computers. PLC-like programming combined with remote I/O hardware, allow a general-purpose desktop computer to overlap some PLCs in certain applications. Regarding the practicality of these desktop computer based logic controllers, it is important to note that they have not been generally accepted in heavy industry because the desktop computers run on less stable operating systems than do PLCs, and because the desktop computer hardware is typically not designed to the same levels of tolerance to temperature, humidity, vibration, and longevity as the processors used in PLCs. In addition to the hardware limitations of desktop based logic, operating systems such as Windows do not lend themselves to deterministic logic execution, with the result that the logic may not always respond to changes in logic state or input status with the extreme consistency in timing as is expected from PLCs. Still, such desktop logic applications find use in less critical situations, such as laboratory automation and use in small facilities where the application is less demanding and critical, because they are generally much less expensive than PLCs.

REFRENCES

[1] W. Bolton, Programmable Logic Controllers, Fifth Edition, Newnes, 2009 ISBN 978-1-85617-751-1, Chapter 1.

[2] Gregory K. McMillan, Douglas M. Considine (ed), Process/Industrial Instruments and Controls Handbook Fifth Edition, McGraw-Hill, 1999 ISBN 0-07-012582-1 Section 3 Controllers.

[3] M. A. Laughton, D. J. Warne (ed), Electrical Engineer's Reference book, 16th edition,Newnes, 2003 Chapter 16 Programmable Controller.

[4] "The father of invention: Dick Morley looks back on the 40th anniversary of the PLC". Manufacturing Automation. 12 September 2008. http://www.automationmag.com/programable-control/features/the-father-of-invention-dick-morley-looks-back-on-the-40th-anniversary-of-the-plc.html.

[5] W. Bolton, Programmable Logic Controllers, Fifth Edition, Newnes, 2009 ISBN 978-1-85617-751-1, Chapter 1.

[6] Daniel Kandray, Programmable Automation Technologies, Industrial Press, 2010 ISBN 978-0-8311-3346-7, Chapter 8 Introduction to Programmable Logic Controller.s